

图书基本信息

书名：<<嵌入式Linux网络体系结构与TCP/IP协议栈>>

13位ISBN编号：9787121129766

10位ISBN编号：7121129760

出版时间：2011-5

出版时间：电子工业出版社

作者：单立平

页数：480

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

内容概要

《嵌入式linux网络体系结构与tcp/ip协议栈》涵盖了linux嵌入式系统开发中网络体系结构实现的主要内容。

《嵌入式linux网络体系结构与tcp/ip协议栈》共分12章，第1章概述linux内核组件与内核技术特点，以及网络体系结构实现应用到的内核开发的基础知识。

第2~5章在介绍了实现网络体系结构、协议栈、设备驱动程序的两个最重要的数据结构sk_buff和net_device的基础上，展示了linux内核中为网络设备驱动程序设计和开发而建立的系统构架，最后以两个实例来具体说明如何着手开发网络设备驱动程序，数据在硬件设备上的接收和发送过程。

第6章讨论了网络协议栈中数据链路层收发数据的设计和实现，以及硬件层与协议层之间的接口。

第7章讲解了网络层ip协议的实现。

第8~9章介绍传输层数据收发过程，重点介绍基于套接字的tcp/udp传输实现。

第10章讨论了linux内核套接字层的实现，以及套接字层与应用层、传输层之间的接口。

第11章介绍网络应用软件的开发技术，以及内核对网络应用的支持。

第12章讲解在嵌入式系统开发中如何将硬件驱动程序、内核代码、应用程序集成在一起下载至芯片中，形成嵌入式可运行的系统，作为全书的总结。

《嵌入式linux网络体系结构与tcp/ip协议栈》可以作为高等院校计算机、通信专业学生学习操作系统的参考书，也可以作为从事嵌入式、计算机行业的工程技术人员的参考书。

作者简介

单立平，电子科技大学计算机学院计算机软件硕士，精通C语言和汇编语言程序设计，以及实时软件系统开发。

在母校执教期间，主讲《计算机组成原理》、《计算机导论》、《汇编语言程序设计》、《C语言程序设计》、《Java程序设计》等课程，参与编写多本高校规划教材，同时从事通信、数据库、嵌入式方向的科研工作。

亦有在信息产业部第十研究所从事卫星跟踪实时软件系统研发的经历。

2005年后定居加拿大，先后在MEADE

WILLIS Company及BlueTree Wireless Data

Inc. 任职，主攻Linux平台下嵌入式无线通信、GPS全球定位系统、汽车电子系统的研究与开发。

书籍目录

第1章 概述

1.1 linux内核组件

1.2 linux内核中的活动

1.2.1 进程和系统调用

1.2.2 硬件中断

1.2.3 tasklet

1.2.4 workqueue

1.2.5 软件中断

1.3 互斥机制

1.3.1 spin lock

1.3.2 读 - 写 spin lock

1.3.3 读 - 复制 - 更新 (read-copy-update , rcu)

1.4 内核模块 (module)

1.4.1 管理内核模块

1.4.2 自动装载模块

1.4.3 模块功能的注册和取消

1.4.4 在模块装载时给模块传递参数

1.4.5 内核和模块的符号表

1.5 内存资源

1.5.1 高速缓冲区 (memory cache)

1.5.2 高速缓存和哈希链表

1.6 时间管理

1.7 嵌入式的挑战

1.8 本章总结

第2章 linux网络包传输的关键数据结构——socket buffer

2.1 socket buffer 设计概述

2.1.1 socket buffer与tcp/ip协议栈

2.1.2 socket buffer的对外接口

2.1.3 socket buffer的特点

2.2 socket buffer的构成

2.2.1 socket buffer的基本组成

2.2.2 socket buffer穿越tcp/ip协议栈

2.3 sk_buff数据域的设计和含义

2.3.1 sk_buff 中的结构管理域

2.3.2 常规数据域

2.3.3 sk_buff 的网络功能配置域

2.4 操作sk_buff的函数

2.4.1 创建和释放socket buffer

2.4.2 数据空间的预留和对齐

2.4.3 复制和克隆

2.4.4 操作队列的函数

2.4.5 引用计数的操作

2.4.6 协议头指针操作

2.5 数据分片和分段

2.5.1 为什么要分割数据包

2.5.2 设计skb_shared_info数据结构的目的是

2.5.3 操作skb_shared_info的函数

2.6 本章总结

第3章 网络设备在内核中的抽象——struct net_device数据结构

3.1 协议栈与网络设备

3.1.1 协议栈软件与网络设备硬件之间的接口

3.1.2 设备独立接口文件dev.c

3.1.3 设备驱动程序

3.1.4 struct net_device数据结构

3.2 struct net_device 数据结构

3.2.1 struct net_device数据结构的数据域

3.2.2 struct net_device数据结构的其他数据域

3.3 struct net_device数据结构中数据域的功能分类

3.3.1 设备管理域

3.3.2 设备配置管理域

3.3.3 设备状态

3.3.4 统计

3.3.5 设备链表

3.3.6 链路层组传送

3.3.7 流量管理

3.3.8 常规域

3.3.9 操作函数结构

3.4 函数指针

3.4.1 设备初始化

3.4.2 传送

3.4.3 硬件协议头

3.4.4 网络统计状态

3.4.5 修改配置

3.5 本章总结

第4章 网络设备在linux内核中识别

4.1 内核初始化的特点

4.1.1 命令行参数

4.1.2 网络子系?的命令行参数

4.2 内核启动过程

4.2.1 用do_initcall函数完成的初始化

4.2.2 标记初始化函数的宏

4.2.3 网络子系统初始化

4.2.4 网络设备的初始化

4.3 网络设备的注册和struct net_device数据结构实例的初始化

4.3.1 初始化函数的任务

4.3.2 网络设备的注册和注销

4.3.3 网络设备的引用计数 (reference count)

4.3.4 允许和禁止网络设备

4.4 网络设备的管理

4.4.1 管理网络设备的链表

4.4.2 网络设备的搜索函数

4.5 事件通知链

- 4.5.1 事件通知链构成
- 4.5.2 注册回调函数到事件通知链
- 4.5.3 通知子系统有事件发生
- 4.5.4 网络子系统的事件通知链
- 4.5.5 网络子系统传送的事件
- 4.6 本章总结
- 第5章 网络设备驱动程序
- 5.1 网络设备驱动程序概述
- 5.1.1 网络设备驱动程序的任务
- 5.1.2 网络设备驱动程序的构成
- 5.2 网络设备与内核的交互
- 5.2.1 设备与内核的交互方式
- 5.2.2 硬件中断
- 5.2.3 中断在内核的实现
- 5.2.4 软件中断
- 5.3 网络设备驱动程序的实现
- 5.3.1 网络适配器的初始化
- 5.3.2 网络设备活动功能函数
- 5.3.3 网络设备管理函数
- 5.3.4 在适配器中支持组发送
- 5.4 cs8900a网络适配器驱动程序的实现
- 5.4.1 cs8900a网络控制芯片的功能概述
- 5.4.2 cs8900a的packetpage结构
- 5.4.3 cs8900a的操作
- 5.4.4 cs8900a设备驱动程序分析
- 5.5 本章总结
- 第6章 数据链路层数据帧的收发
- 6.1 关键数据结构
- 6.1.1 struct napi_struct数据结构
- 6.1.2 struct softnet_data数据结构
- 6.2 数据帧的接收处理
- 6.2.1 napi的实现
- 6.2.2 netif_rx函数分析
- 6.3 网络接收软件中断
- 6.3.1 net_rx_action的工作流程
- 6.3.2 net_rx_action函数的实现细节
- 6.3.3 从输入队列中读取数据帧
- 6.3.4 处理输入数据帧
- 6.4 数据链路层?网络层的接口
- 6.4.1 输入数据帧协议解析
- 6.4.2 实现数据链路层与网络层接口的关键数据结构
- 6.4.3 接口的组织
- 6.5 数据链路层对数据帧发送的处理
- 6.5.1 启动/停止设备发送数据
- 6.5.2 调度设备发送数据帧
- 6.5.3 队列策略接口
- 6.5.4 dev_queue_xmit函数

- 6.5.5 发送软件中断
- 6.5.6 watchdog时钟
- 6.6 本章总结
- 第7章 网络层传送
 - 7.1 internet协议的基本概念
 - 7.1.1 internet协议的任务
 - 7.1.2 internet 协议头
 - 7.1.3 linux内核中描述ip协议头的数据结构
 - 7.2 ip协议实现前的准备工作
 - 7.2.1 协议初始化
 - 7.2.2 与网络过滤子系统的交互
 - 7.2.3 与路由子系统的交互
 - 7.3 输入数据包在ip层的处理
 - 7.3.1 ip_rcv函数分析
 - 7.3.2 ip_rcv_finish函数分析
 - 7.3.3 接收操作中ip选项的处理
 - 7.4 ip选项
 - 7.4.1 ip选项的格式
 - 7.4.2 描述ip选项的数据结构
 - 7.4.3 linux内核对ip选项的处理
 - 7.4.4 linux内核对ip选项处理的具体实现
 - 7.5 ipv4数据包的前送和本地发送
 - 7.5.1 数据包的前送
 - 7.5.2 dst_output函数的实现
 - 7.5.3 本地发送的处理
 - 7.6 在ip层的发送
 - 7.6.1 执行发送的关键函数
 - 7.6.2 发送数据包相关信息的数据结构
 - 7.6.3 ip_queue_xmit函数
 - 7.6.4 ip_append_data函数预备
 - 7.6.5 ip_append_data函数分析
 - 7.6.6 ip_append_page 函数
 - 7.6.7 ip_push_pending_frames函数
 - 7.6.8 发送数据包的整体过程
 - 7.7 与相邻子系统的接口
 - 7.8 数据包的分片与重组
 - 7.8.1 数据分片需要考虑的问题
 - 7.8.2 在上层分片的效率
 - 7.8.3 数据包分片/重组使用的ip 协议头数据域
 - 7.9 本章总结
- 第8章 传输层udp协议的实现
 - 8.1 udp协议基础
 - 8.2 udp协议实现的关键数据结构
 - 8.2.1 udp协议头的数据结构
 - 8.2.2 udp的控制缓冲区
 - 8.2.3 udp套接字的数据结构
 - 8.2.4 应用程序发送给udp负载数据的数据结构

8.3 udp、套接字层、ip层之间的接口

8.3.1 udp协议实例与套接字层间的接口

8.3.2 udp协议与ip层之间的接口

8.4 发送udp数据报的实现

8.4.1 初始化一个连接

8.4.2 在udp套接字上发送数据包

8.4.3 向ip层发送数据包

8.4.4 从用户地址空间复制数据到数据报

8.5 udp 协议接收的实现

8.5.1 udp协议接收的处理函数

8.5.2 将数据包放入套接字接收队列的处理函数

8.5.3 udp协议接收广播与组发送数据包

8.5.4 udp的哈希链表

8.5.5 将数据包放到套接字接收队列

8.6 udp协议在套接字层的接收处理

8.6.1 函数输入参数

8.6.2 函数处理流程

8.7 本章总结

第9章 传输层tcp协议的实现

9.1 cp协议简介

9.1.1 cp是可靠协议

9.1.2 tcp是面向连接的协议

9.1.3 tcp是按字节流交换的协议

9.1.4 tcp协议实现的功能

9.2 描述tcp协议实现的关键数据结构

9.2.1 tcp协议头数据结构

9.2.2 tcp的控制缓冲区

9.2.3 tcp套接字的数据结构

9.2.4 tcp协议选项options

9.2.5 应用层传送给传输层信息的数据结构

9.3 在tcp协议、套接字、ip层之间的接口

9.3.1 管理套接字与tcp接口的数据结构

9.3.2 初始化套接字与传输层之间的接口

9.3.3 tcp与ip层之间的接收接口

9.3.4 tcp与ip层之间的发送接口

9.3.5 初始化tcp 套接字

9.4 tcp协议实例接收过程的实现

9.4.1 tcp_v4_rcv函数的实现

9.4.2 fast path和prequeue队列的处理

9.4.3 处理tcp的blocklog 队列

9.4.4 套接字层的接收函数

9.5 linux 内核中tcp发送功能的实现

9.5.1 将数据从用户地址空间复制到内核socket buffer

9.5.2 tcp 数据段输出

9.5.3 发送过程的状态机

9.6 tcp套接字的连接管理

9.6.1 tcp连接初始化

- 9.6.2 tcp状态从closed切换到syn_sent
- 9.6.3 tcp连接的状态管理
- 9.6.4 tcp连接为established状态时的接收处理
- 9.6.5 tcp的time_wait状态处理
- 9.7 本章总结
- 第10章 套接字层实现
 - 10.1 套接字概述
 - 10.1.1 什么是套接字
 - 10.1.2 套接字与管理套接字的数据结构
 - 10.1.3 套接字与文件
 - 10.2 套接字层的初始化
 - 10.3 地址族的值和协议交换表
 - 10.3.1 协议交换表的数据结构
 - 10.3.2 套接字支持多协议栈的实现
 - 10.4 ipv4中协议成员注册和初始化
 - 10.5 套接字api系统调用的实现
 - 10.5.1 系统调用简述
 - 10.5.2 套接字api系统调用的实现
 - 10.6 创建套接字
 - 10.6.1 sock_create函数创建套接字
 - 10.6.2 协议族套接字创建函数的管理
 - 10.6.3 af_inet套接字的创建
 - 10.7 i/o系统调用和套接字
 - 10.8 本章总结
- 第11章 应用层——网络应用套接字编程
 - 11.1 套接字描述符
 - 11.1.1 family参数
 - 11.1.2 type 参数
 - 11.1.3 protocol 参数
 - 11.1.4 af_xxx与pf_xxx形式的常数
 - 11.2 地址格式
 - 11.2.1 字节顺序
 - 11.2.2 地址结构
 - 11.2.3 支持地址格式转换的函数
 - 11.2.4 获取网络配置信息
 - 11.2.5 编程示例
 - 11.2.6 将地址与套接字绑定
 - 11.3 套接字连接
 - 11.3.1 connect函数分析
 - 11.3.2 服务器套接字建立侦听队列
 - 11.3.3 建立套接字连接
 - 11.4 数据的传送
 - 11.4.1 send函数
 - 11.4.2 传送数据的函数
 - 11.4.3 接收数据的函数
 - 11.4.4 recvfrom、recvmsg函数
 - 11.4.5 编程示例

11.5 套接字选项

11.5.1 设置套接字选项

11.5.2 读取套接字选项

11.6 out-of-band 数据

11.7 非阻塞和异步i/o操作

11.8 本章总结

第12章 嵌入式系统网络应用技术

12.1 嵌入式系统的设计要素

12.2 嵌入式系统开发环境的构成

12.2.1 硬件构成

12.2.2 典型的硬件开发环境

12.2.3 软件交叉平台开发环境

12.2.4 嵌入式软件的开发步骤

12.3 将网络设备驱动程序加入内核

12.3.1 配置新网络设备

12.3.2 编译新驱动程序

12.4 内核配置

12.4.1 目标硬件及内核、库配置

12.4.2 内核组件配置

12.4.3 应用配置

12.5 集成应用程序并下载至目标板

12.5.1 集成应用程序

12.5.2 将执行文件下载至目标板

12.6 本章总结

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>